ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

(Laboratorio di)
Amministrazione di sistemi

Servizi rete infrastrutturali

Marco Prandini

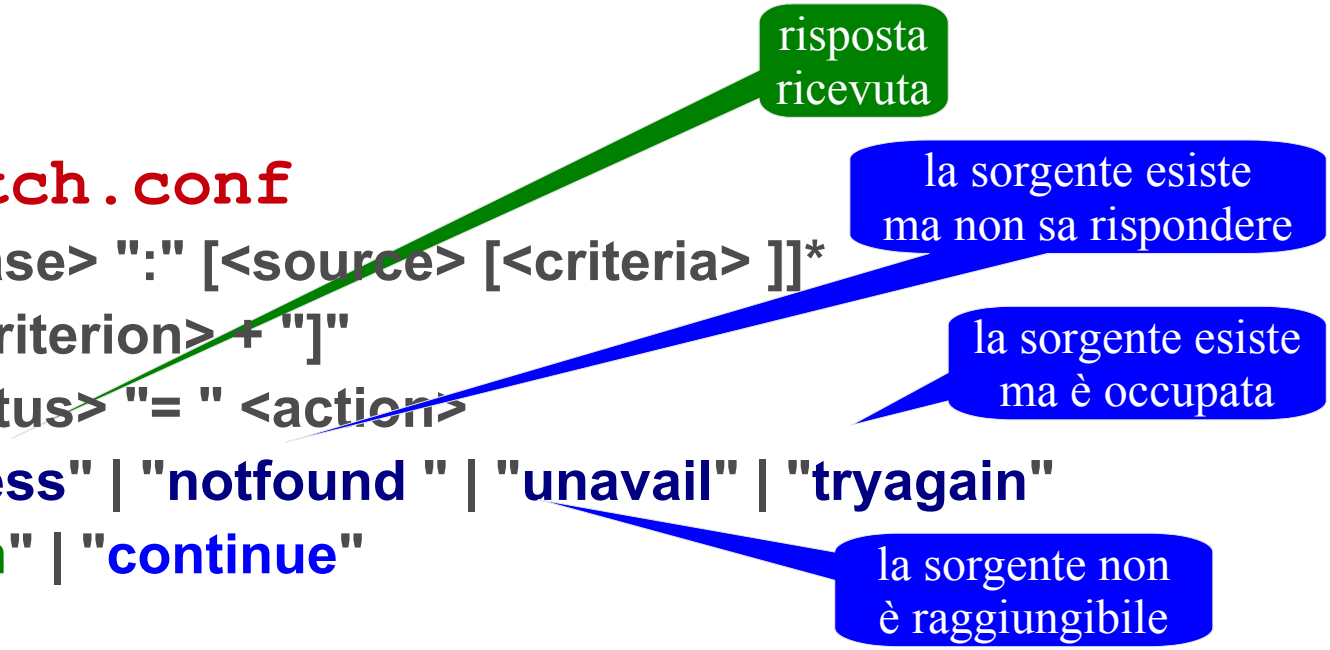
Dipartimento di Informatica – Scienza e Ingegneria

Risoluzione dei nomi - generalità

- La mappatura da nomi di host a indirizzi IP e viceversa è uno dei tanti casi in cui il sistema ha bisogno di un dizionario di nomi
- Il primo accorgimento adottato da GNU/Linux riguarda la *scelta della sorgente di informazioni*
  - *Name Service Switch*
  - svolta dalla libreria C di sistema
  - supporta un set fisso di possibili database
  - configurata tramite `/etc/nsswitch.conf`
    - vedi man page omonima

NSS

- Sintassi di `nsswitch.conf`
  - <entry> ::= <database> ":" [<source> [<criteria>]]*
  - <criteria> ::= "[" <criterion> + "]"
  - <criterion> ::= <status> "= " <action>
  - <status> ::= "**success**" | "**notfound** " | "**unavail**" | "**tryagain**"
  - <action> ::= "**return**" | "**continue**"



(i colori indicano l'azione di default)

- Es.

```
passwd: files nis ldap
group: files ldap
hosts: ldap [NOTFOUND=return] dns files
```

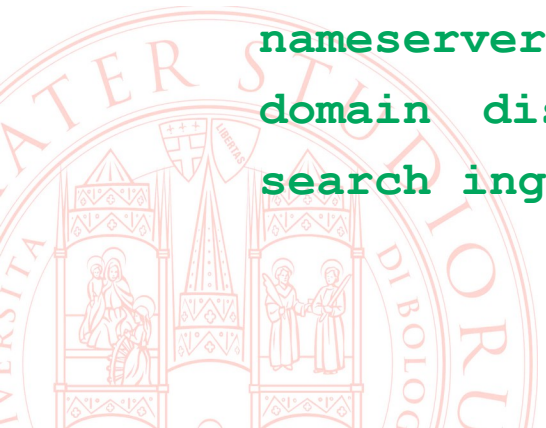
Risoluzione dei nomi – host e IP

- Es: `hosts: ldap [NOTFOUND=return] dns files`

- `files` → la sorgente di informazioni è il file `/etc/hosts`
  - formato: `<IP> <FQDN> [<ALIAS> …]`

    esempio: `8.8.8.8  dns.google.com  gdns`
- `dns` → la sorgente di informazioni è il sistema DNS
  - l'interrogazione di server DNS è un'ulteriore set di funzioni della libreria C di sistema, il *resolver*
  - si configura attraverso `/etc/resolv.conf`
  - esempio

```
nameserver 137.204.58.1
domain  disi.unibo.it
search ing.unibo.it
```



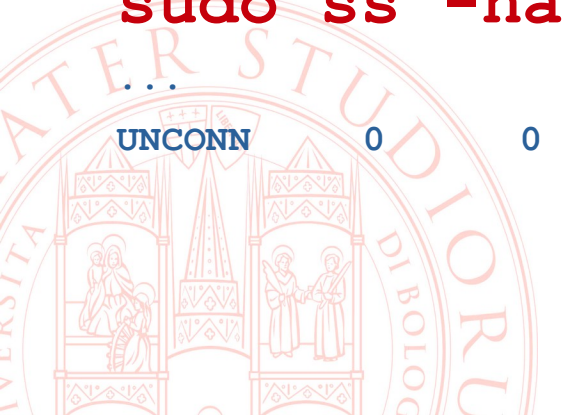
DNS caching

- Spesso si trova un server DNS locale
  - Miglioramento prestazioni
  - Maggiore flessibilità per contesti dinamici

```
~$ cat /etc/resolv.conf
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.1.1
```

  - Tutti gli IP che iniziano per 127 puntano a localhost

```
sudo ss -naup | grep 127.0.1.1:53
...
UNCONN     0      0      127.0.1.1:53     *:*     users:(("dnsmasq",pid=2154,fd=4))
```



Risoluzione di nomi via NSS

- Il comando getent permette di interrogare i database del name service switch

```
getent <db name> <keyword>
```

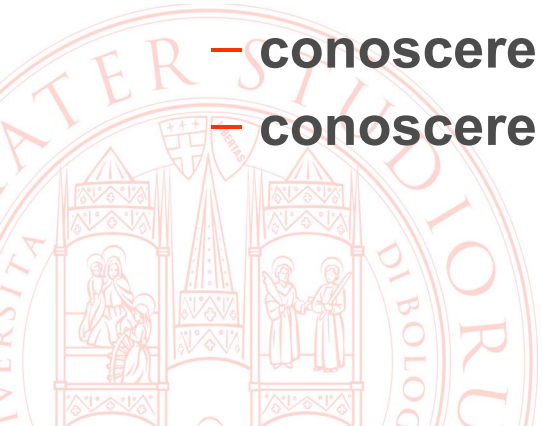
- Esempi:

```
$ getent passwd las
las:x:1000:1000:Lab Amministrazione Sistemi,,,:/home/las:/bin/bash

$ getent hosts www.unibo.it
137.204.24.35   atrproxy.unibo.it www.unibo.it
```

Risoluzione nomi DNS diretta

- Per interrogare direttamente il DNS e avere più controllo sulle query si usano tipicamente host e dig
  - non considerano nsswitch
  - usano i nameserver di resolv.conf di default
  - possono interrogare un server specifico
- `host` (tipicamente per conversioni IP ⟷ nome)
  - query di un nome: `host www.unibo.it`
  - query a un server specifico: `host www.unibo.it 8.8.8.8`
- `dig` (tipicamente per ottenere informazioni legate a un dominio diverse da nomi host)
  - conoscere i Mail eXchanger: `dig mx example.com`
  - conoscere i Name Server: `dig ns example.com`

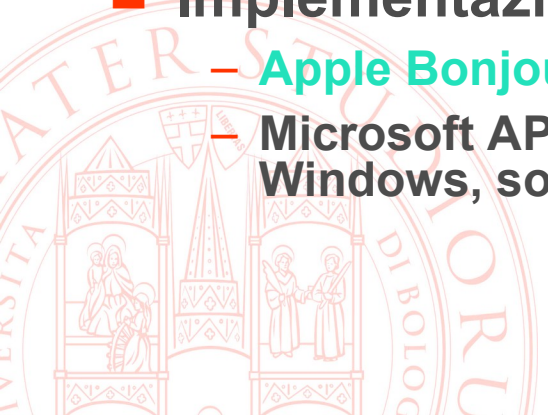


zeroconf

- Lo Zeroconf Working Group dell'IETF si è occupato di standardizzare varie soluzioni presenti sul mercato per completare il quadro della configurazione automatica

  http://www.zeroconf.org/

- Non considera la standardizzazione delle comunicazioni a livello applicativo con periferiche, come fa UPnP, ma i layer comuni a tutti
  - **link-local addressing** per determinare automaticamente un indirizzo di rete
  - **multicast DNS** per la traduzione di nomi in indirizzi in assenza di un DNS unicast configurato manualmente
  - **service discovery** basato su server DNS aggiornabile dinamicamente per registrare servizi
- Implementazioni comuni:
  - Apple Bonjour (mDNS e SD)
  - Microsoft APIPA (solo link-local addressing; mDNS e SD via Bonjour for Windows, soluzioni native da Windows 10)



Link-local addressing

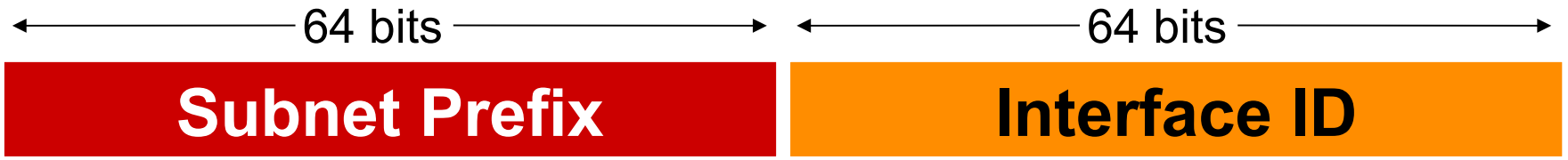
- Gli indirizzi layer 2 per definizione valgono solo localmente (link local)
  - sono garantiti univoci sulla LAN ma non globalmente
  - non sono instradabili
- Idea: adottare lo stesso approccio per il layer 3 per rendere possibile la “generazione spontanea” di subnet
- Due metodi distinti per IPv4 e IPv6

Link local IPv4

- RFC 3927 Dynamic Configuration of IPv4 Link-Local Addresses
  - riserva a questo scopo la classe 169.254/16
  - propone una serie di best practice per delimitare l'uso degli indirizzi link local
    - non devono essere assegnati a interfacce che hanno indirizzi instradabili
    - non devono essere distribuiti via DHCP
    - non devono essere stabilmente associati a nomi DNS
  - lettura interessante per tutti i problemi che possono sorgere!
- Meccanismo di assegnazione
  - entra in gioco solo se l'interfaccia non ha già un indirizzo assegnato staticamente o via DHCP
  - scelta IP random nel range 169.254.1.0 – 169.254.254.255
    - con seme legato a caratteristica univoca (es. MAC address)
      - riduce probabilità di conflitto
      - tendenzialmente risulta in ri-assegnamenti stabili
  - verifica che l'IP sia disponibile via ARP probe
  - annuncio di acquisizione via gratuitous ARP

Link local IPv6

- Indirizzi IPv6 (RFC 4291 e aggiornamenti)

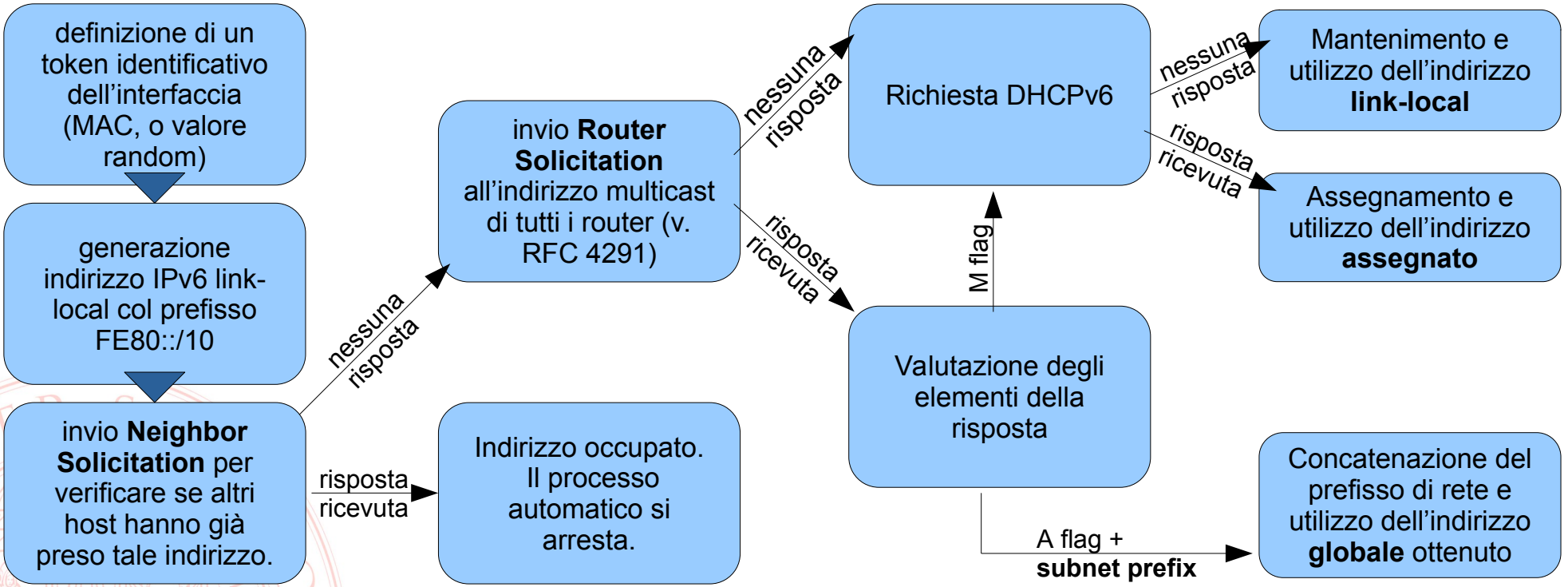


- IPv6 definisce un proprio range di indirizzi link-local
  - logicamente equivalente a 169.254/16
  - prefisso FE80::/10
  - ogni interfaccia può costruire un proprio IPv6 link-local con questo schema:

1. MAC: `00:10:a4:01:23:45`
2. EUI-64: `00:10:a4:ff:fe:01:23:45`
3. EUI-64 first byte: `00000000`
4. Il settimo bit viene invertito: `00000010`
5. Modified EUI-64: `02:10:a4:ff:fe:01:23:45`
6. IPv6: `fe80:0000:0000:0000:0210:a4ff:fe01:2345`
   (aka `fe80::0210:a4ff:fe01:2345`)

SLAAC

- IPv6 non utilizza ARP ma ha sistemi più complessi e flessibili per individuare indirizzi liberi o utilizzati e determinare se la rete locale è anche raggiungibile dall'esterno
- StateLess Address AutoConfiguration (RFC 4862 e aggiornamenti) è un algoritmo per costruire indirizzi link-local se possibile validi globalmente



mDNS

- L'associazione nomi-indirizzi viene gestita secondo la RFC 6762 Multicast DNS
  - definisce che il TLD .local. sia riservato a host appartenenti a una rete link-local
  - impone che richieste di risoluzione per nomi che terminano in .local. siano inviate all'indirizzo link-local di multicast 224.0.0.251 (IPv4) o FF02::FB (IPv6) porta 5353
    - utilizzabile anche per risolvere altri domini, volendo
    - idem per il reverse mapping di 254.169.in-addr.arpa. , 8.e.f.ip6.arpa. , 9.e.f.ip6.arpa. , a.e.f.ip6.arpa. , b.e.f.ip6.arpa.
  - raccomanda di strutturare i nomi in modo flat per i record A e AAAA, mentre lascia libere le etichette per gli altri record
    - senza implicare gerarchie o deleghe
  - prevede la possibilità di mantenere aperte le connessioni dopo aver ricevuto richieste, per inviare automaticamente aggiornamenti su tutte le nuove risorse registrate o modificate

DNS-SD

- La rilevazione automatica di servizi disponibili in rete segue la RFC 6763 DNS-Based Service Discovery
- Stabilisce un formato di entry DNS per descrivere la collocazione in rete, i protocolli applicativi ed eventuali parametri da utilizzare
- Es.

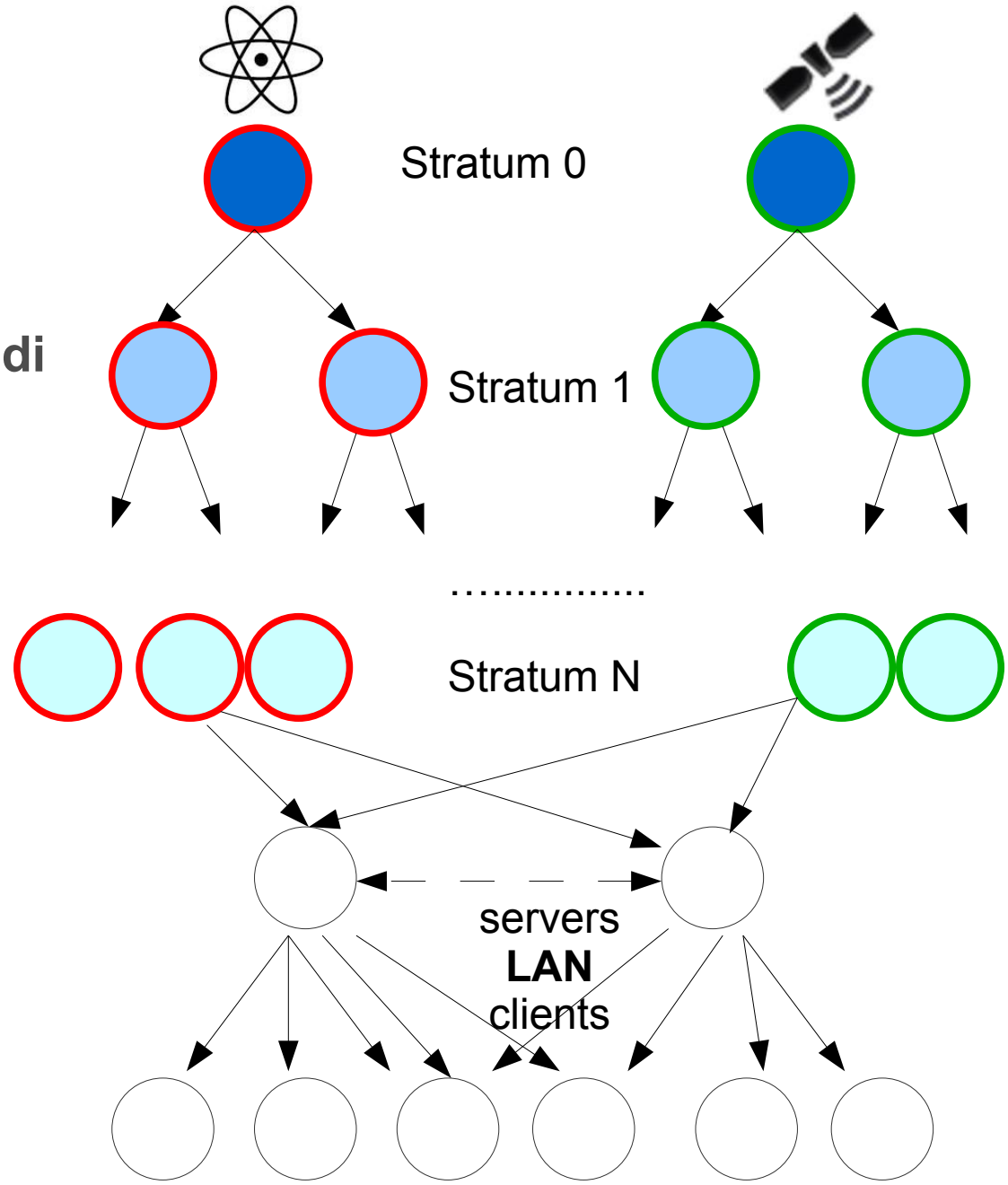
```
b._dns-sd._udp  IN PTR @    ;  b = browse domain
lb._dns-sd._udp IN PTR @    ; lb = legacy browse domain
_http._tcp  PTR New\ Employee\ Information\ Page._http._tcp
New\ Employee\ Information\ Page._http._tcp     SRV 0 0 80 ne-info
                                                TXT path=/
```

Sincronizzazione

- L'allineamento dell'ora di un sistema ad un orologio di riferimento è cruciale
  - per la diagnostica dei problemi (timestamp su log)
  - per i protocolli di autenticazione e autorizzazione (i messaggi hanno una vita limitata)
  - per la sincronizzazione di azioni distribuite
  - per il valore legale di azioni compiute attraverso i computer
- È possibile usare un protocollo che compensa i ritardi di rete per ottenere informazioni precise via Internet: *Network Time Protocol* (NTP)
  - sito ufficiale: https://www.ntp.org/
  - grande quantità di informazioni su: https://www.eecis.udel.edu/~mills/ntp.html

NTP in breve

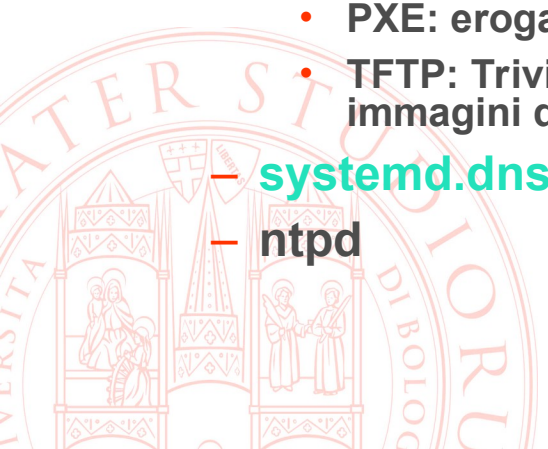
- Preciso
  - poche decine di millisecondi di scarto su WAN
  - <1 millisecondo su LAN
  - supporto di sorgenti HW (oscillatori, GPS, ...)
- Standard
  - RFC 5905
  - portato su ogni architettura nota
- Scalabile e affidabile
  - *multi-server*
  - *strata*
  - *peering*
  - *auto-keying*



Linux tools

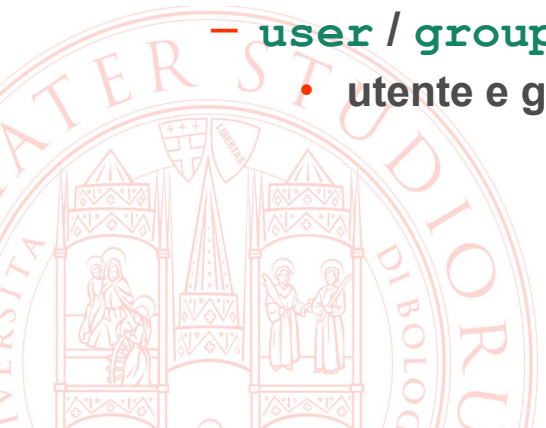
- Diverse possibilità
- Client side:
  - Avahi (link-local, mDNS, DNS-SD)
  - ISC dhcp
  - systemd.network (ogni possibile modo di configurare la rete, incluso link-local)
  - systemd-resolved (DNS/mDNS resolver, sostituisce resolvconf)
  - ntpd / ntpdate
- Server side
  - dnsmasq (DHCP, DNS)

    e inoltre:
    - PXE: erogazione di Pre-boot eXecution Environment per l'avvio di sistemi diskless
    - TFTP: Trivial FTP, utilizzato dall'ambiente PXE per il caricamento da remoto di immagini di sistema
  - systemd.dnssd (DNS-SD)
  - ntpd



Server side - dnsmasq

- Su reti di piccole dimensioni dnsmasq è una scelta pratica per fornire i servizi necessari all'avvio zeroconf
- Configurazione generale
  - file predefinito: `/etc/dnsmasq.conf`
- Opzioni base
  - `bind-interfaces`
    - evita conflitti in caso si vogliano usare più istanze di dnsmasq per diverse reti connesse al server
  - `interface=<interface name>`
  - `listen-address=<ipaddr>`
    - mettono dnsmasq in ascolto solo sull'interfaccia o l'indirizzo specificati (anche più di una/uno)
  - `user` / `group` / `pid`
    - utente e gruppo UNIX del processo, file in cui salvare il PID



DHCP con dnsmasq – configurazione base

- il server DHCP è disabilitato per default, se non sono specificate le opzioni descritte (in forma semplificata) di seguito
- `dhcp-range=<start-addr>[,<end-addr>|<mode>][,<netmask>[,<broadcast>]][,<lease time>]`
  - può essere specificata più volte per servire diverse subnet
  - fornisce indirizzi tra `<start-addr>` e `<end-addr>`
  - se specificato, imposta il `<lease time>`
  - la `<netmask>` è opzionale per reti direttamente connesse al server
  - al posto di `<end addr>`, `<mode>` può essere `static` per abilitare il server sulla rete specificata senza servire indirizzi dinamici, ma solo quelli specificati con l'opzione `dhcp-host`
- `dhcp-host=[<hwaddr>][,<ipaddr>][,<hostname>][,<lease_time>][,ignore]`
  - assegna `<hostname>`, `<ipaddr>` e `<lease time>` stabili all'host con MAC=`<hwaddr>`
  - con `ignore` non offrirà mai un lease all'host specificato
- `dhcp-hostfile` permette di specificare (una directory di) file contenenti informazioni formattate come la parte a destra dell'= di `dhcp-host`

DHCP con dnsmasq – opzioni

- `dhcp-option=[<opt>|option:<opt-name>|option6:<opt>|option6:<opt-name>],[<value>[,<value>]]`
  - uso più comune: `dhcp-option=<opt>,<value>`
- Opzioni definite nella RFC 2132
  - comuni:

1	netmask
2	fuso orario (time offset rispetto a UTC)
3	default gateway
4	time server
6	DNS server
12	host name
15	domain name
121	static route (parametro:network/netmask,gateway)

DNS con dnsmasq

- Opzioni base
  - `port=<dns server port>`
    - di default `53`, se impostato a `0` disabilita il server DNS
  - `local-service`
    - accetta query DNS solo dagli host delle subnet locali al server
    - ha effetto solo se non è specificata nessuna delle opzioni
      - `interface`
      - `except-interface`
      - `listen-address`
      - `auth-server options`
    - tipica impostazione di default
      - utile in rete locale
      - previene DNS amplification attacks

DNS con dnsmasq – risoluzione

- non è un resolver ricorsivo, deve appoggiarsi a uno esterno
- di default prende gli indirizzi dei nameserver upstream da `/etc/resolv.conf`
  - plug-in su sistemi che sono già configurati
  - file aggiornato dinamicamente da demoni dhcp, ppp, ecc.
    - dnsmasq si accorge automaticamente delle modifiche
- spesso però lo si vuole usare anche localmente
  - vantaggio: caching
  - in questo caso /etc/resolv.conf deve contenere `nameserver 127.0.0.1`
- i server upstream possono (devono per uso locale) essere specificati
  - con `resolv-file=<file>`
    - sopprime l'uso di /etc/resolv.conf
  - con `server=[/<domain>/]<ipaddr>`
    - per evitare l'uso di /etc/resolv.conf si deve aggiungere `no-resolv`

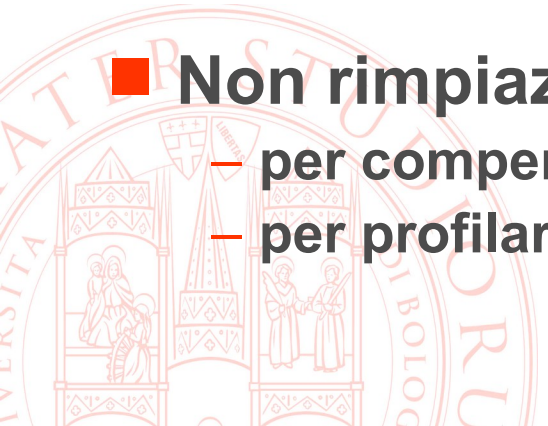
NTP su Linux

- Il demone *ntpd* è client e/o server in funzione della configurazione
- `/etc/ntp.conf` – esempio

```
server 0.ubuntu.pool.ntp.org
server 1.ubuntu.pool.ntp.org
peer fellow.server.lan
# By default, exchange time with everybody, but don't allow
configuration.
restrict -4 default kod notrap nomodify nopeer noquery
restrict -6 default kod notrap nomodify nopeer noquery
# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1
restrict ::1
```

NTP – inizializzazione e uso sporadico

- Il tool `ntpdate` permette di sincronizzare l'orologio locale a un server NTP
  - senza parametri usa i server in `ntp.conf`
    - `ntpd` non deve essere attivo
  - accetta come parametro un server specifico
- L'ora viene modificata in due modi
  - se la differenza è più di 0.5 secondi: **step**
  - se la differenza è meno di 0.5 secondi: **slew** con `adjtime()`
- Non rimpiazza **ntpd**, che usa algoritmi sofisticati
  - per compensare errori e ritardi dei pacchetti dai server
  - per profilare il comportamento dell'orologio locale



Lato client – DHCP

- Il pacchetto **isc-dhcp-client** fornisce il comando `dhclient` (v. man page)
- Lanciato senza parametri avvia un demone che tenta di configurare tutte le interfacce
- Tipicamente avviato da **interfaces** con
  - `auto <ifname>`
  - `iface <ifname> inet dhcp`
- Parametri impostabili in `/etc/dhcp/dhclient.conf`
- Hook per l'esecuzione automatica di script al cambio di stato dell'interfaccia
  - `/etc/dhcp/dhclient-enter-hooks.d/*`
  - `/etc/dhcp/dhclient-exit-hooks.d/*`

Lato client – zeroconf

- Il framework avahi può fornire
  - uno stack completo mDNS/DNS-SD con API per l'integrazione delle funzionalità in qualsiasi programma C
  - un demone per gestire le registrazioni di nuovi servizi in modo orchestrato da qualsiasi programma non scritto in C, via D-Bus
  - un client/wrapper C che semplifica l'utilizzo di D-Bus
  - adattatori per integrare le API di avahi negli event loop dei sistemi grafici come GNOME e KDE
- Il demone è responsabile ad esempio della scoperta automatica di stampanti in una rete locale
- Sono disponibili pacchetti con strumenti per svolgere funzioni singole specifiche

Lato client – link local

- Il pacchetto **avahi-autoipd** fornisce il comando omonimo
  - implementa IPv4 Link Local
  - demone indipentente, oppure
  - … nel file **interfaces**:
  - `auto <ifname>`
  - `iface <ifname> inet ipv4all`
  - ad ogni cambio di stato dell'interfaccia invoca `/etc/avahi/avahi-autoipd.action`
- Può essere usato come fallback se DHCP fallisce
  - plugin per **dhclient** (hook nelle directory specificate)